Projekt z NLP 2020/2021 Bioinformatyka

Dane: 50 tys. recenzji filmowych podzielonych na dwie kategorie: pozytywne i negatywne.

- Proszę wykonać poniższe kroki każdorazowo komentując sens wykonania kolejnych działań.
- Rozważam 2 możliwości oddania projektu: jako skrypt ipynb/html lub pdf z wklejonym kodem/wynikami.

1. Preprocessing

- odpowiednie przygotowanie i oczyszczenie tekstu do dalszej analizy

- czy dane są zbalansowane?

2. Wstępna analiza tekstu

- ile wystąpiło wszystkich słów a ile unikatowych? (we wszystkich recenzjach)

- narysuj rozkład występowania 40 najpopularniejszych słów (niebędących znakami interpunkcyjnymi czy stopwordsami)

3. Analiza danych za pomocą klasycznych algorytmów nauczania maszynowego i reprezentacji Set of Words.

Proszę podzielić dane na dwie grupy: treningowe (80%) i testowe (20%) a następnie zbudować modele oparte o następujące podejścia:

- Naive Bayes

- Regresja logistyczna

- SVM

- klasyfikator zbiorczy oparty o trzy powyższe metody

Wyznacz średnią dokładność dla każdej z metod (dla każdej metody zbuduj model 10-krotnie). Jakie było 15 najbardziej decydujących słów w modelu Naive Bayes (dla przykładowego modelu)?

4. Analiza tekstu w oparciu o sieci neuronowe

Zbuduj:

- prostą sieć opartą o reprezentację Bag of Words i jedno przekształcenie liniowe. Tutaj analogicznie jak w pkt.3 dzielimy dane na dwie grupy - treningowe [80%] i testowe [20%]

- sieć neuronową wykorzystującą embeddingi 200D wytrenowane na Wikipedia 2014 (GloVe). Każda recenzja reprezentowana jest jako suma bądź średnia arytmetyczna embeddingów w nią wchodzących.

- sieć neuronową opartą o embeddingi 200D oraz LSTM lub/oraz GRU.

5. Kilka zdań podsumowania.

Uwagi:

W dwóch ostatnich podpunktach proszę zastosować różne zestawy parametrów (w tym rodzaje przekształceń, funkcje aktywacji, liczba epok, learning rate; ewentualnie: optymalizator) aby uzyskać możliwie najlepszą dokładność na zbiorze testowym. Dane powinny być podzielone na trzy grupy - dane treningowe, testowe i walidacyjne w proporcji 70%/15%/15% odpowiednio. Wybrać model, który sprawdza się najlepiej na zbiorze walidacyjnym (w zadanym zakresie epok). Rozważ różne rozmiary batch'a. Etapy uczenia zaprezentuj graficznie.

Przy wykorzystaniu LSTM/GRU proszę pamiętać o kwestii batchowania w sieciach rekurencyjnych (recenzje zazwyczaj są różnej długości, a z drugiej strony wymagany jest tensor o zadanej długości). Możliwe rozwiązania: pad_sequence, TBatcher, zastosowanie BucketIterator. Ten ostatni ma następującą składnie:

train_iter = torchtext.data.BucketIterator(train, batch_size=32, sort_key=lambda x: len(x.txt), sort_within_batch=True, repeat=False).